**Project3 회고**

2019-15915 유형욱

Multicore 프로그래밍을 진행하며 가장 어려웠던 부분, 가장 집중했던 부분은 메모리 Control 관련 부분이었다. 크게 큰 수에서 오는 문제와, thread 관리에서 오는 문제로 구분할 수 있을 것이다.

**큰 수에서 오는 문제**

1조라는 int형이 감당할 수 없는 크기의 수를 다루다 보니, 100억 이상으로 수를 키워 진행했을 때 예상과 다른 결과가 나오는 경우가 있었고, 한 동안 그 원인을 찾지 못했다. 그 후 int64\_t로 바꾸어 할당했을 때 그 문제를 해결할 수 있었으나, runtime이 늘어나게 되었다. 프로그래밍 시 10억을 기준으로 프로그래밍을 짜고, 시험해본 후, 나중에 숫자를 1조로 바꾸는 방식을 사용했는데, 1조가 물리적으로 커 소수의 개수를 새는 알고리즘을 여러 차례 개선해야 하기도 하였다. 그 외 초기에는 vector를 사용했던 것을 array로 바꾸고 int형에서 char형으로 바꾸는 등 최대한 빠른 프로그램을 만드는 데 집중하게 되었다.

**Thread 관리 문제**

Thread의 개수를 크게 늘리면 Segmentation Fault가 나타나 적절한 Thread의 개수를 찾기 위해 노력하였다. 결국 계산의 유용성을 갖춘 100개의 Thread를 사용하였다. 무엇보다 여러 스레드에서 한 자료에 동시에 접근할 때 생기는 문제를 해결하는 것이 가장 어려운 부분이었다. 포인터를 사용하여 속도를 개선한 알고리즘을 만들었었지만, 여러 Thread에서 생성한 포인터가 같은 자료를 중복으로 할당해 계산 상의 문제가 발생해 폐기하기도 하였다.